Simplex Architecture Meets RTLola

Bernd Finkbeiner*, Jessica Schmidt†, Maximilian Schwenger*
Department of Computer Science
Saarland University
Saarbrücken, Germany
*{finkbeiner, schwenger}@react.uni-saarland.de
†schmidt-jessica@stud.uni-saarland.de

Abstract

Designing controllers for safety-critical cyber-physical systems is a challenging task due to their complex dynamics and only partial access to information. Despite these difficulties, machine learned controllers show remarkable success. Their outstanding performance is tarnished by an opaque structure that prohibits reasoning about their internals. A remedy for this problem is the Simplex architecture. It embeds an arbitrarily complex controller into a verifiable structure that monitors controller decisions. Upon detection of potentially harmful commands, the architecture falls back to a simple and safe controller. While validation of control decisions is easier than finding them, it still has to account for complex temporal dependencies. At the same time, deployment in embedded safety-critical system requires the monitor to be formally verifiable and to cope with strict resource limitations. In this talk we will discuss the monitoring module of the Simplex architecture on the example of an artificial pancreas and propose using the RTLola monitoring framework.

Modern medicine confronts doctors with vast amounts of patient data. In addition to information assessed during consultations, wearable or implantable devices in particular produce a continuous stream of real-time data. Based on this, the doctor tailors a treatment specific to the patient's health status. Consider, for example, an implanted, automated insulin pump, i.e. an artificial pancreas. It continuously measures the glucose concentration in the patient's tissue and administers insulin if the concentration exceeds the threshold to hyperglycemia. Insulin is a hormone that — in a nutshell — regulates the absorption of glucose into liver, fat, and muscle cells. Thus, it effectively reduces the glucose level in blood. Since abnormally low blood sugar is dangerous, and may lead to fainting and even death, the insulin dosage must be regulated very carefully. This, however, depends on a multitude of factors such as the patient's reaction to both insulin and high glucose levels, or their eating and exercise habits. As a result, configuring the insulin pump properly is far from simple and neither is the validation of a configuration. Recent endeavors strive to use machine learned controllers for artificial pancreata [1]. While they perform remarkably well, they are even harder to understand than configurations provided by doctors.

However, given the potential serious implications on the patient's health, there is a strong interest in making the health monitoring process more explainable. Better explainability is also a prerequisite for the certification of such devices. In this work, we propose the automated generation of the monitors out of a high-level, easily understandable language with a formal semantics.

Our approach is based on the Simplex architecture [2], which allows for benefitting from highly performant, opaque solutions in form of a *Complex Controller (CC)* while retaining a certifiable, safe *Base Controller (BC)*. An intermediate module, the *Decision Module (DM)* receives the same information as the controllers and validates decisions of the CC. Upon detection of spurious or potentially harmful decisions, it transfers control to the BC. In the case of artificial pancreata, the BC usually refrains from administering any insulin. While this might result in overly high glucose levels, which is harmful in the long run, it does not put the patient at immediate risk. The general architecture is depicted in Figure 1.

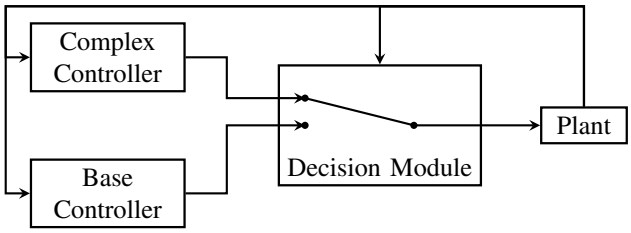

Figure 1: Basic idea of the Simplex architecture used for monitoring safety-critical cyber-physical systems.

In principle, the decision module can be very simple, for example based on thresholds on the patient's glucose level and the previously administered dose of insulin. More sophisticated monitoring can, however, help to identify

```
input glucose: Float64
input insulin: Float64
output gradient: Float64 := glucose − glucose.offset(by: −1).defaults(to: glucose)
trigger glucose < c_hypo ∧ insulin > 0.0
    "Administration of insulin despite hypoglycemia."
trigger gradient.aggr(over: 30min, using: avg) < c_drop ∧ insulin > 0.0
    "Administration of insulin despite downwards trend."
trigger gradient.aggr(over: 10min, using: avg) > c_spike
    ∧ insulin.aggr(over: 10min, using: ∑) = 0.0 ∧ glucose > c_hyper
    "Warning: Spike in glucose not counter−acted by insulin."
```

Figure 2: An RTLola specification for monitoring a controller of an artificial pancreas. It checks whether the controller administers insulin despite a trend towards hypoglycemia and warns if the controller does not react on a spike in the glucose level.

potential risks more accurately. Criteria might be based on the timing of decisions and reactions, as well as real-time properties concerning the gradient of the glucose level when exposed to insulin.

In choosing a language for the definition of the monitor, a fundamental challenge is that the language needs to be sufficiently expressive to capture the properties, while still offering a certifiable solution, which involves verifiability and understandability. While logics satisfy the latter as they are sufficiently well understood, they often lack capabilities to express quantified real-time properties. Contrarily, programming languages allow for specifying essentially arbitrary properties, yet their analysis poses a hard problem.

In addition to this trade-off, the monitor is required to cope with strict limits on resource consumption. Especially in embedded devices such as an implantable artificial pancreas, the available computing power and working memory is strictly bounded before deployment. To guarantee the safe operation, it its therefore mandatory to bound the memory usage of the monitor statically.

For these reasons we propose using the RTLola [3] framework. It evolves around a stream-based real-time specification language with the same name. A specification in this language can be realized in hardware [4]. Static analyzes reveal the amount of memory required and provide insights into the running time of the monitor.

Example: Consider an RTLola monitor as decision module in an implanted artificial pancreas with a specification shown in Figure 2. The specification declare two inputs, the decision of the CC and the currently measured glucose level. Note that the decision of the BC is considered correct and thus does not need to be monitored. The output stream `gradient` computes the discrete change between two consecutive measurements of the current glucose level. The specification then checks three properties. The first two are: Does the controller attempt to administer insulin even though a) the glucose level is already too low, b) the insulin level is decreasing strongly over the last 30 minutes. Lastly, the third trigger checks whether the artificial pancreas does not deliver insulin despite a spike in the glucose level that already lead to exceeding the threshold to hyperglycemia.

Conclusion: The decision module in the Simplex architecture plays a significant role in the development of safety-critical systems such as medical implants. Since it is subject to a variety of constraints, a suitable solution for the generation of these modules is necessary, for which we propose RTLola. In the talk, we want to discuss the requirements on the decision module and directions for future research in this field.

References

[1] M. K. Bothe, L. Dickens, K. Reichel, A. Tellmann, B. Ellger, M. Westphal, and A. A. Faisal, "The use of reinforcement learning algorithms to meet the challenges of an artificial pancreas," *Expert Review of Medical Devices*, vol. 10, no. 5, pp. 661–673, 2013. [Online]. Available: https://doi.org/10.1586/17434440.2013.827515

[2] L. Sha, "Using simplicity to control complexity," *IEEE Software*, vol. 18, no. 4, pp. 20–28, 2001. [Online]. Available: https://doi.org/10.1109/MS.2001.936213

[3] P. Faymonville, B. Finkbeiner, M. Schledjewski, M. Schwenger, M. Stenger, L. Tentrup, and H. Torfah, "Streamlab: Stream-based monitoring of cyber-physical systems," in *CAV*, ser. LNCS, I. Dillig and S. Tasiran, Eds., vol. 11561. Springer, 2019, pp. 421–431. [Online]. Available: https://doi.org/10.1007/978-3-030-25540-4_24

[4] J. Baumeister, B. Finkbeiner, M. Schwenger, and H. Torfah, "FPGA stream-monitoring of real-time properties," *ACM Trans. Embedded Comput. Syst.*, vol. 18, no. 5s, pp. 88:1–88:24, 2019. [Online]. Available: https://doi.org/10.1145/3358220